QUBIC Tutorial

Yu Zhang
Juan Xie
Qin Ma

Gene expression data is very important in experimental molecular biology (Brazma and Vilo 2000), especially for cancer study (Fehrmann et al. 2015). The large-scale microarray data and RNA-seq data provide good opportunity to do the gene co-expression analyses and identify co-expressed gene modules; and the effective and efficient algorithms are needed to implement such analysis. Substantial efforts have been made in this field, such as Cheng and Church (2000), Plaid (Lazzeroni, Owen, and others 2002), Bayesian Biclustering (BCC, Gu and Liu 2008), among them Cheng and Church and Plaid has the R package implementation. It is worth noting that our in-house biclustering algorithm, QUBIC (Li et al. 2009), is reviewed as one of the best programs in terms of their prediction performance on benchmark datasets. Most importantly, it is reviewed as the best one for large-scale real biological data (Eren et al. 2012).

Until now, QUBIC has been cited over 110 times (via Google Scholar) and its web server, QServer (Zhou et al. 2012), was developed in 2012 to facilitate the users without comprehensive computational background (Zhou et al. 2012). In the past five years, the cost of RNA-sequencing decreased dramatically, and the amount of gene expression data keeps increasing. Upon requests from users and collaborators, we developed this R package of QUBIC to void submitting large data to a webserver.

The unique features of our R package include (1) updated and more stable back-end resource code (re-written by C++), which has better memory control and is more efficient than the one published in 2009. For an input dataset in Arabidopsis, with 25,698 genes and 208 samples, we observed more than 40% time saving; and (2) comprehensive functions and examples, including discretize function, heatmap drawing and network analysis.

Other languages

If R is not your thing, there is also a C version of `QUBIC`.

Help

If you are having trouble with this R package, contact the maintainer, Yu Zhang.

Install and load

Stable version from BioConductor

```
source("https://bioconductor.org/biocLite.R")
biocLite("QUBIC")
```

Or development version from GitHub

```
install.packages("devtools")
devtools::install_github("zy26/QUBIC")
```

Load `QUBIC`

```
library("QUBIC")
```

Functions

There are six functions provided by QUBIC package.

- `qudiscretize()`creates a discrete matrix for a given gene expression matrix;
- `BCQU()`performs a qualitative biclustering for real matrix;
- `BCQUD()`performs a qualitative biclustering for discretized matrix;
- `quheatmap()`can draw heatmap for singe bicluster or overlapped biclusters;
- `qunetwork()` can automatically create co-expression networks based on the identified biclusters by QUBIC;
- `qunet2xml()`can convert the constructed co-expression networks into XGMML format for further network analysis in Cytoscape, Biomax and JNets. The following examples illustrate how these functions work.

Example of a random matrix with two diferent embedded biclusters

```
library(QUBIC)
set.seed(1)
# Create a random matrix
test <- matrix(rnorm(10000), 100, 100)
colnames(test) <- paste("cond", 1:100, sep = "_")
rownames(test) <- paste("gene", 1:100, sep = "_")

# Discretization
matrix1 <- test[1:7, 1:4]
matrix1
```

```
##            cond_1      cond_2     cond_3      cond_4
## gene_1 -0.6264538 -0.62036668  0.4094018  0.89367370
## gene_2  0.1836433  0.04211587  1.6888733 -1.04729815
## gene_3 -0.8356286 -0.91092165  1.5865884  1.97133739
## gene_4  1.5952808  0.15802877 -0.3309078 -0.38363211
## gene_5  0.3295078 -0.65458464 -2.2852355  1.65414530
## gene_6 -0.8204684  1.76728727  2.4976616  1.51221269
## gene_7  0.4874291  0.71670748  0.6670662  0.08296573
```

```
matrix2 <- qudiscretize(matrix1)
matrix2
```

```
##        cond_1 cond_2 cond_3 cond_4
## gene_1     -1      0      0      1
## gene_2      0      0      1     -1
## gene_3      0     -1      0      1
## gene_4      1      0      0     -1
## gene_5      0      0     -1      1
## gene_6     -1      0      1      0
## gene_7      0      1      0     -1
```

```r
# Fill bicluster blocks
t1 <- runif(10, 0.8, 1)
t2 <- runif(10, 0.8, 1) * (-1)
t3 <- runif(10, 0.8, 1) * sample(c(-1, 1), 10, replace = TRUE)
test[11:20, 11:20] <- t(rep(t1, 10) * rnorm(100, 3, 0.3))
test[31:40, 31:40] <- t(rep(t2, 10) * rnorm(100, 3, 0.3))
test[51:60, 51:60] <- t(rep(t3, 10) * rnorm(100, 3, 0.3))

# QUBIC
res <- biclust::biclust(test, method = BCQU())
summary(res)
```

```
##
## An object of class Biclust
##
## call:
##  biclust::biclust(x = test, method = BCQU())
##
## Number of Clusters found:  39
##
## Cluster sizes:
##                    BC 1 BC 2 BC 3 BC 4 BC 5 BC 6 BC 7 BC 8 BC 9 BC 10
## Number of Rows:      10    9    9    9   10    5    3    2    3     3
## Number of Columns:    9    9    8    7    5    3    5    6    4     4
##                    BC 11 BC 12 BC 13 BC 14 BC 15 BC 16 BC 17 BC 18 BC 19
## Number of Rows:        3     2     2     2     2     2     2     2     2
## Number of Columns:     4     6     6     6     6     6     6     6     6
##                    BC 20 BC 21 BC 22 BC 23 BC 24 BC 25 BC 26 BC 27 BC 28
## Number of Rows:        2     2     2     2     2     2     2     2     2
## Number of Columns:     5     5     5     5     5     5     5     5     5
##                    BC 29 BC 30 BC 31 BC 32 BC 33 BC 34 BC 35 BC 36 BC 37
## Number of Rows:        2     2     2     2     2     2     2     2     2
## Number of Columns:     5     5     5     5     5     5     5     5     5
##                    BC 38 BC 39
## Number of Rows:        2     2
## Number of Columns:     5     5
```

```r
# Show heatmap
hmcols <- colorRampPalette(rev(c("#D73027", "#FC8D59", "#FEE090", "#FFFFBF",
    "#E0F3F8", "#91BFDB", "#4575B4")))(100)
# Specify colors

par(mar = c(4, 5, 3, 5) + 0.1)
quheatmap(test, res, number = c(1, 3), col = hmcols, showlabel = TRUE)
```

```
## [1] "yto 0"
## [1] "xlo 0"
```

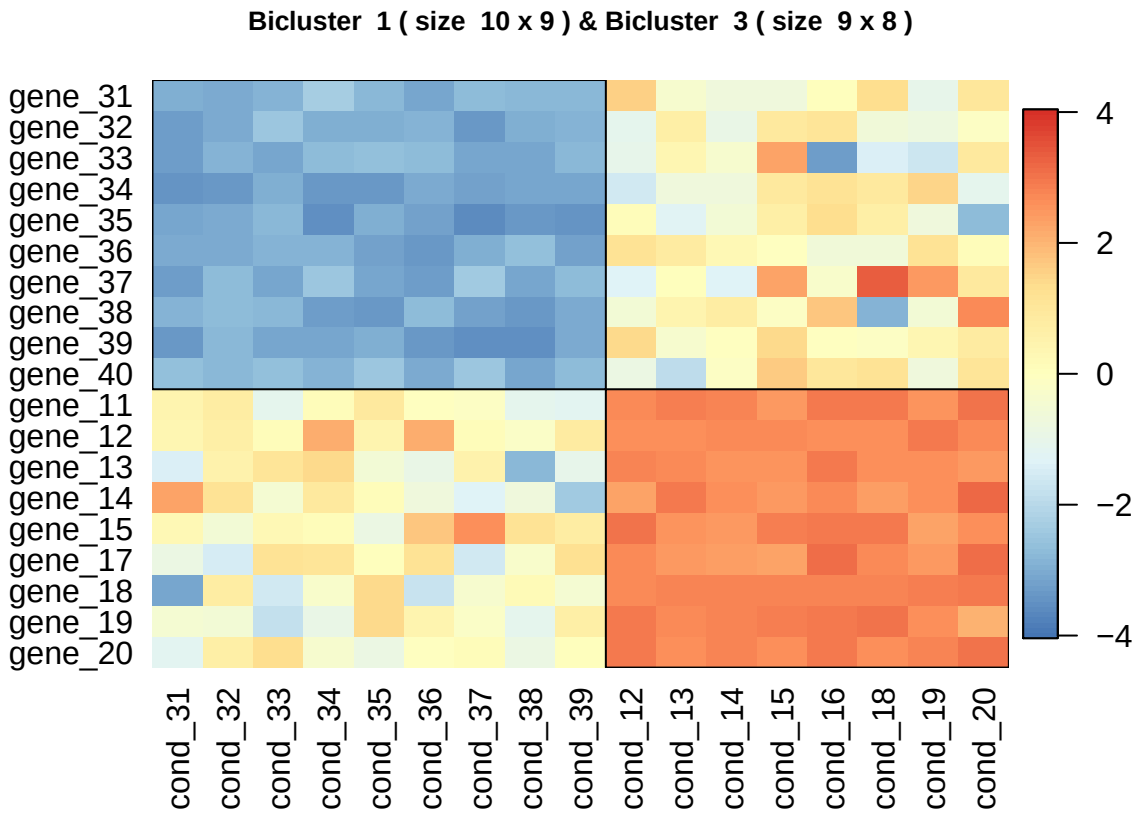

Figure 1: Heatmap for two overlapped biclusters in the simulated matrix

Example of BicatYeast

```
library(QUBIC)
data(BicatYeast)

# Discretization
matrix1 <- BicatYeast[1:7, 1:4]
matrix1
```

```
##           cold_green_6h cold_green_24h cold_roots_6h cold_roots_24h
## 249364_at    -0.2759300     -0.5108508    1.74476670     2.12442300
## 253423_at    -0.9405282      3.2669048   -0.37776557     0.06860917
## 250327_at     1.6419950      1.4484175    0.33474782    -0.15095752
## 247474_at     0.6903505      1.6705408    0.04386528    -0.45295456
## 252661_at     1.7493315      0.9260773    2.05519100     2.11200260
## 258239_at     0.6110116     -0.6083303    0.60419910     0.43582130
## 248910_at     1.4501406     -0.3107802    0.16640233     0.37186486
```

```
matrix2 <- qudiscretize(matrix1)
matrix2
```

```
##           cold_green_6h cold_green_24h cold_roots_6h cold_roots_24h
## 249364_at             0             -1             0              1
```

```
## 253423_at            -1              1             0              0
## 250327_at             1              0             0             -1
## 247474_at             0              1             0             -1
## 252661_at             0             -1             0              1
## 258239_at             1             -1             0              0
## 248910_at             1             -1             0              0
```

```
# QUBIC
x <- BicatYeast
system.time(res <- biclust::biclust(x, method = BCQU()))
```

```
##    user  system elapsed
##   0.113   0.000   0.098
```

```
summary(res)
```

```
##
## An object of class Biclust
##
## call:
##  biclust::biclust(x = x, method = BCQU())
##
## Number of Clusters found:  77
##
## Cluster sizes:
##                     BC 1 BC 2 BC 3 BC 4 BC 5 BC 6 BC 7 BC 8 BC 9 BC 10
## Number of Rows:       72   53   37   80   56   98   47   26   45    35
## Number of Columns:     4    5    7    3    4    2    4    7    4     5
##                     BC 11 BC 12 BC 13 BC 14 BC 15 BC 16 BC 17 BC 18 BC 19
## Number of Rows:        29    34    42    33    41    41    32    53    39
## Number of Columns:      6     5     4     5     4     4     5     3     4
##                     BC 20 BC 21 BC 22 BC 23 BC 24 BC 25 BC 26 BC 27 BC 28
## Number of Rows:        38    74    29    47    23    27    22    33    41
## Number of Columns:      4     2     5     3     6     5     6     4     3
##                     BC 29 BC 30 BC 31 BC 32 BC 33 BC 34 BC 35 BC 36 BC 37
## Number of Rows:        40    19    51    20    16    32    31    23    30
## Number of Columns:      3     6     2     5     6     3     3     4     3
##                     BC 38 BC 39 BC 40 BC 41 BC 42 BC 43 BC 44 BC 45 BC 46
## Number of Rows:        44    44    41    13    39    26    39    19    25
## Number of Columns:      2     2     2     6     2     3     2     4     3
##                     BC 47 BC 48 BC 49 BC 50 BC 51 BC 52 BC 53 BC 54 BC 55
## Number of Rows:        25    25    25     8    36     9     8    12    18
## Number of Columns:      3     3     3     9     2     8     9     6     4
##                     BC 56 BC 57 BC 58 BC 59 BC 60 BC 61 BC 62 BC 63 BC 64
## Number of Rows:        23    16    32    31    20    28    14    11    18
## Number of Columns:      3     4     2     2     3     2     4     5     3
##                     BC 65 BC 66 BC 67 BC 68 BC 69 BC 70 BC 71 BC 72 BC 73
## Number of Rows:        18    17    24    24     9    14    14    20    12
## Number of Columns:      3     3     2     2     5     3     3     2     3
##                     BC 74 BC 75 BC 76 BC 77
## Number of Rows:         9    17     8     3
## Number of Columns:      4     2     3     5
```

We can draw heatmap for single bicluster.

```
# Draw heatmap for the 2th bicluster identified in BicatYeast data

library(RColorBrewer)
paleta <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
par(mar = c(5, 4, 3, 5) + 0.1, mgp = c(0, 1, 0), cex.lab = 1.1, cex.axis = 0.5,
    cex.main = 1.1)
quheatmap(x, res, number = 2, showlabel = TRUE, col = paleta)
```

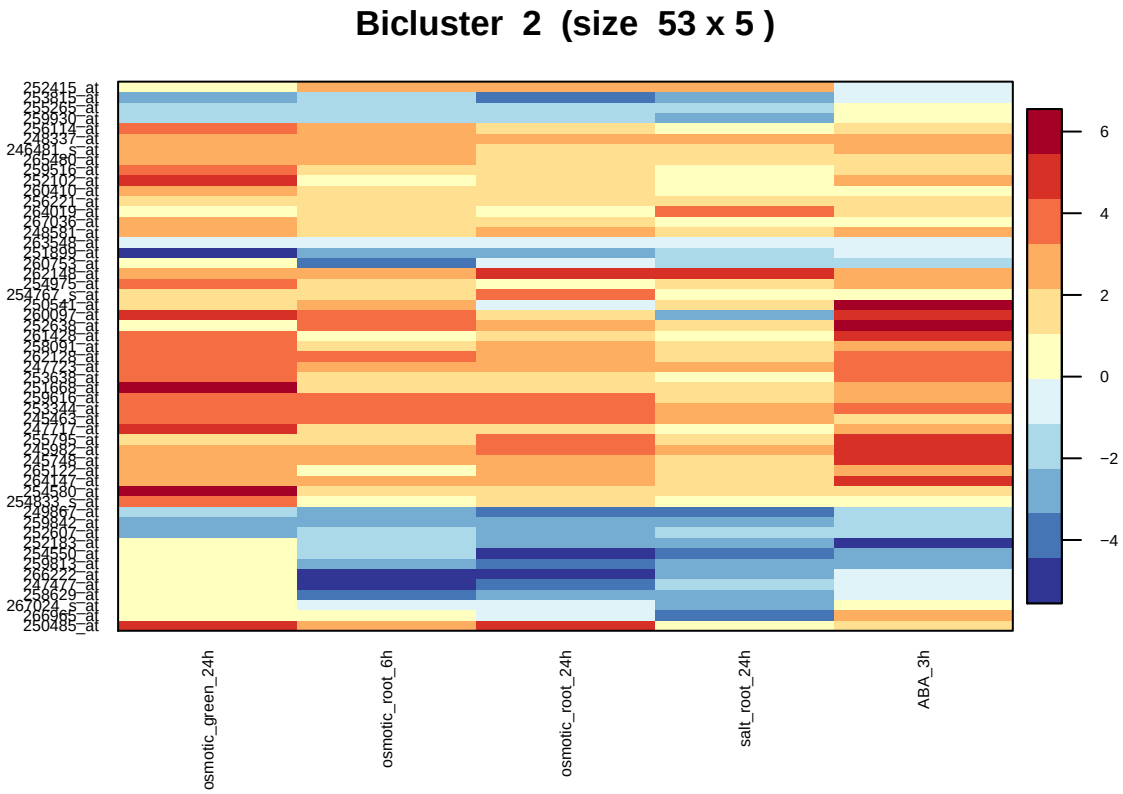

Figure 2: Heatmap for the second bicluster identified in the BicatYeast data. The bicluster consists of 53 genes and 5 conditions

We can draw heatmap for overlapped biclusters.

```
# Draw for the 2th and 3th biclusters identified in BicatYeast data

par(mar = c(5, 5, 5, 5), cex.lab = 1.1, cex.axis = 0.5, cex.main = 1.1)
paleta <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
quheatmap(x, res, number = c(2, 3), showlabel = TRUE, col = paleta)
```

```
## [1] "yto 0"
## [1] "xlo 0"
```

We can draw network for single bicluster.

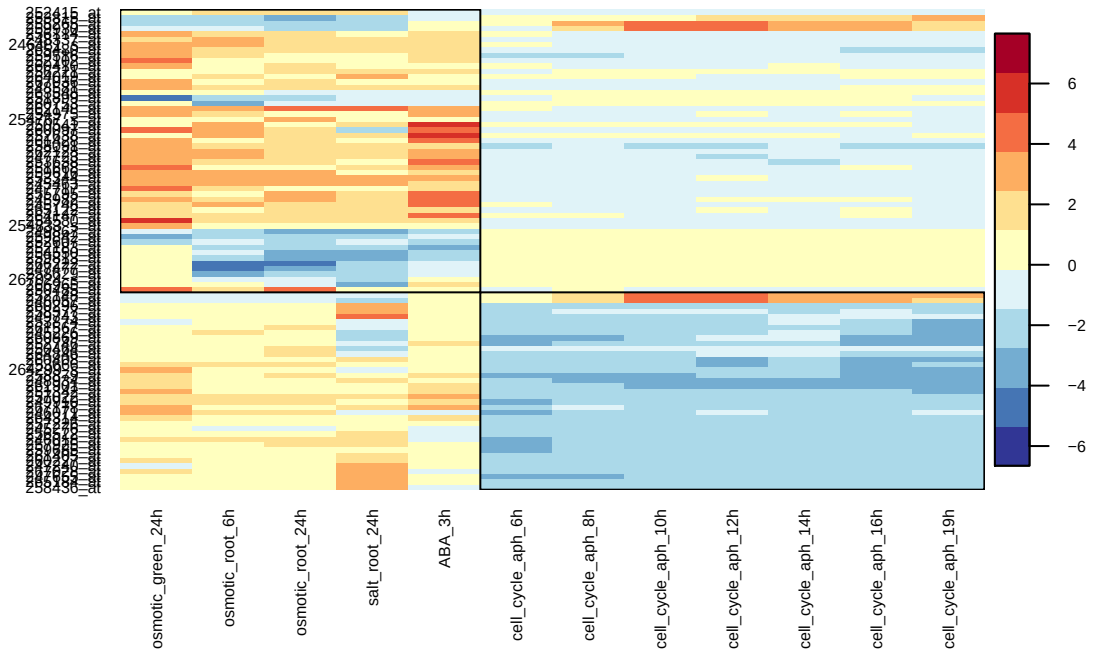

Figure 3: Heatmap for the second and third biclusters identified in the BicatYeast data. Bicluster #2 (topleft) consists of 53 genes and 5 conditions, and bicluster #3 (bottom right) consists of 37 genes and 7 conditions.

```
# Construct the network for the 2th identified bicluster in BicatYeast
net <- qunetwork(x, res, number = 2, group = 2, method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
    qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", minimum = 0.6,
        color = cbind(rainbow(length(net[[2]]) -  1), "gray"), edge.label = FALSE)
```

We can also draw network for overlapped biclusters.

```
net <- qunetwork(x, res, number = c(2, 3), group = c(2, 3), method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
    qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", minimum = 0.6,
        legend.cex = 0.5, color = c("red", "blue", "gold", "gray"), edge.label = FALSE)
```

```
# Output overlapping heatmap XML, could be used in other software such
# as Cytoscape, Biomax or JNets
sink('tempnetworkresult.gr')
qunet2xml(net, minimum = 0.6, color = cbind(rainbow(length(net[[2]]) - 1), "gray"))
sink()
# We can use Cytoscape, Biomax or JNets open file named 'tempnetworkresult.gr'
```

Example of E.coli data

The E.coli data consists of 4,297 genes and 466 conditions.

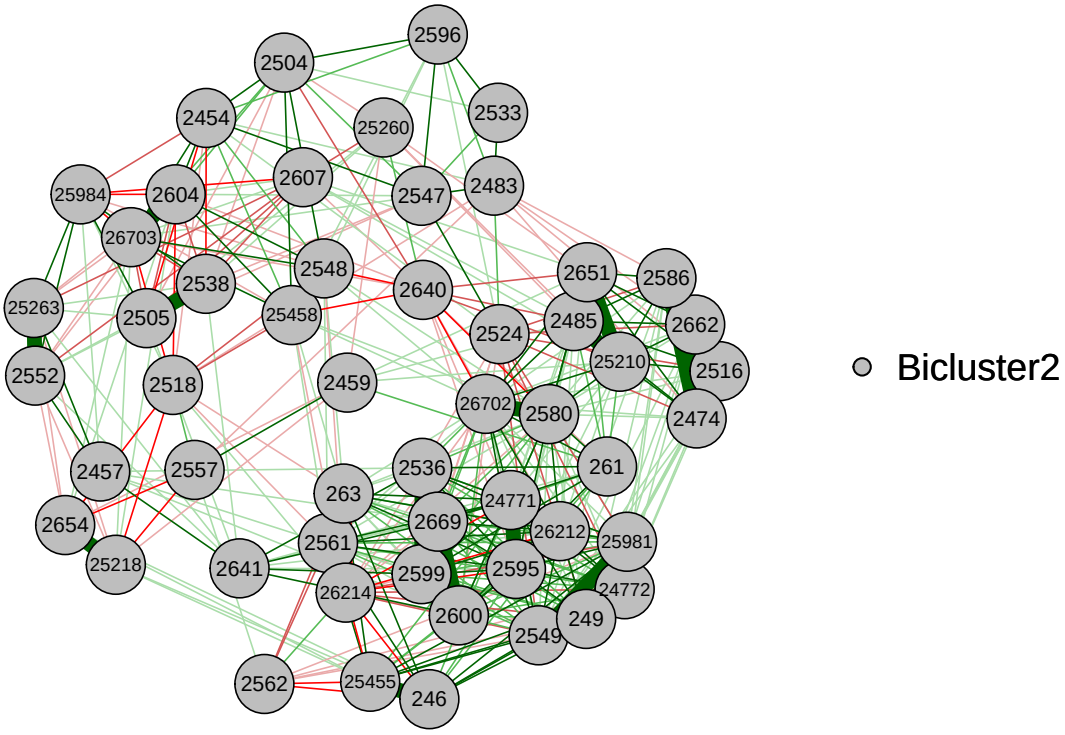

Figure 4: Network for the second bicluster identified in the BicatYeast data.

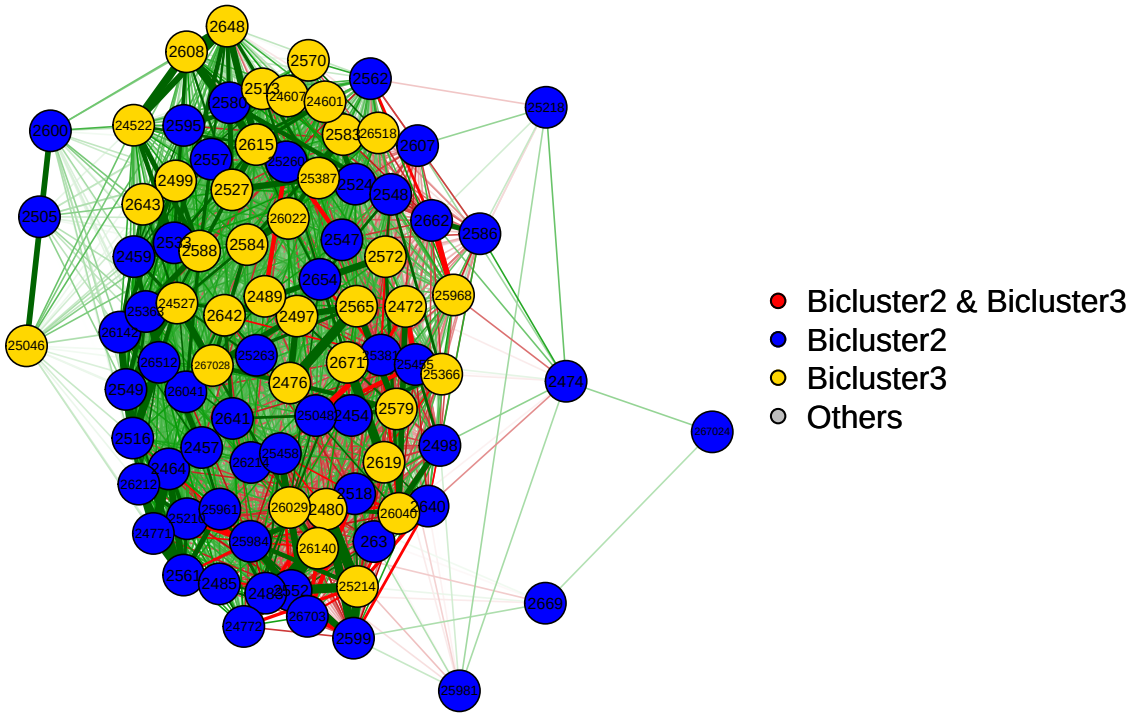

Figure 5: Network for the second and third biclusters identified in the BicatYeast data.

```
library(QUBIC)

# Load E.coli data
if (requireNamespace("QUBICdata", quietly = TRUE)) {
    data("ecoli", package = "QUBICdata")
} else {
    warning("Could not load QUBICdata. Bioconductor version <= 3.2? ", call. = FALSE)
    print("Downloading fallback file.")
    x <- read.table(
        "https://www.sdstate.edu/ps/research/bioinformatics/upload/ecoli_466_4297.txt",
        header = TRUE)

    # Convert data frame to matrix
    ecoli <- as.matrix(x[, -1])
    rownames(ecoli) <- x[, 1]
}

# Discretization
matrix1 <- ecoli[1:7, 1:4]
matrix1
```

```
##       dinI_U_N0025 dinP_U_N0025 lexA_U_N0025 lon_U_N0025
## b4634     9.077693     9.225537     9.138900    9.114353
## b3241     7.122300     7.195453     7.051193    7.124200
## b3240     7.184417     7.336610     7.283377    7.188263
## b3243     7.902090     7.963167     7.847747    7.943650
## b3242     6.801900     6.843213     6.795007    6.889897
## b2836     9.114207     9.133303     9.167487    9.189480
## b0885     9.057120     8.918723     8.985483    9.002663
```

```
matrix2 <- qudiscretize(matrix1)
matrix2
```

```
##       dinI_U_N0025 dinP_U_N0025 lexA_U_N0025 lon_U_N0025
## b4634           -1            1           0           0
## b3241            0            1          -1           0
## b3240           -1            1           0           0
## b3243            0            1          -1           0
## b3242            0            0          -1           1
## b2836           -1            0           0           1
## b0885            1           -1           0           0
```

```
# QUBIC
res <- biclust::biclust(ecoli, method = BCQU(), r = 1, q = 0.06, c = 0.95, o = 100,
    f = 0.25, k = max(ncol(ecoli)%/%20, 2))
system.time(res <- biclust::biclust(ecoli, method = BCQU(), r = 1, q = 0.06, c = 0.95,
    o = 100, f = 0.25, k = max(ncol(ecoli)%/%20, 2)))
```

```
##    user  system elapsed
##  11.298   0.037   6.331
```

```
summary(res)
```

```
##
## An object of class Biclust
##
## call:
##  biclust::biclust(x = ecoli, method = BCQU(), r = 1, q = 0.06,
##      c = 0.95, o = 100, f = 0.25, k = max(ncol(ecoli)%/%20,
##          2))
##
## Number of Clusters found:  20
##
## Cluster sizes:
##                     BC 1 BC 2 BC 3 BC 4 BC 5 BC 6 BC 7 BC 8 BC 9 BC 10
## Number of Rows:      437  121   51  108  103   65   41   26   27    20
## Number of Columns:    29   45   94   44   38   38   31   33   31    27
##                     BC 11 BC 12 BC 13 BC 14 BC 15 BC 16 BC 17 BC 18 BC 19
## Number of Rows:        25    23    17    18    14    15    13    11     5
## Number of Columns:     19    20    23    21    26    20    22    25    32
##                     BC 20
## Number of Rows:         6
## Number of Columns:     23
```

```
# Draw heatmap for the 5th bicluster identified in E.coli data

library(RColorBrewer)
paleta <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
par(mar = c(5, 4, 3, 5) + 0.1, mgp = c(0, 1, 0), cex.lab = 1.1, cex.axis = 0.5,
    cex.main = 1.1)
quheatmap(ecoli, res, number = 5, showlabel = TRUE, col = paleta)
```

```
library(RColorBrewer)
paleta <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
par(mar = c(5, 4, 3, 5), cex.lab = 1.1, cex.axis = 0.5, cex.main = 1.1)
quheatmap(ecoli, res, number = c(4, 8), showlabel = TRUE, col = paleta)
```

```
## [1] "yto 0"
## [1] "xlo 1"
```

```
# construct the network for the 5th identified bicluster in E.coli data
net <- qunetwork(ecoli, res, number = 5, group = 5, method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
    qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", minimum = 0.6,
        color = cbind(rainbow(length(net[[2]]) - 1), "gray"), edge.label = FALSE)
```

```
# construct the network for the 4th and 8th identified bicluster in E.coli data
net <- qunetwork(ecoli, res, number = c(4, 8), group = c(4, 8), method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
    qgraph::qgraph(net[[1]], groups = net[[2]], legend.cex = 0.5, layout = "spring",
        minimum = 0.6, color = c("red", "blue", "gold", "gray"), edge.label = FALSE)
```

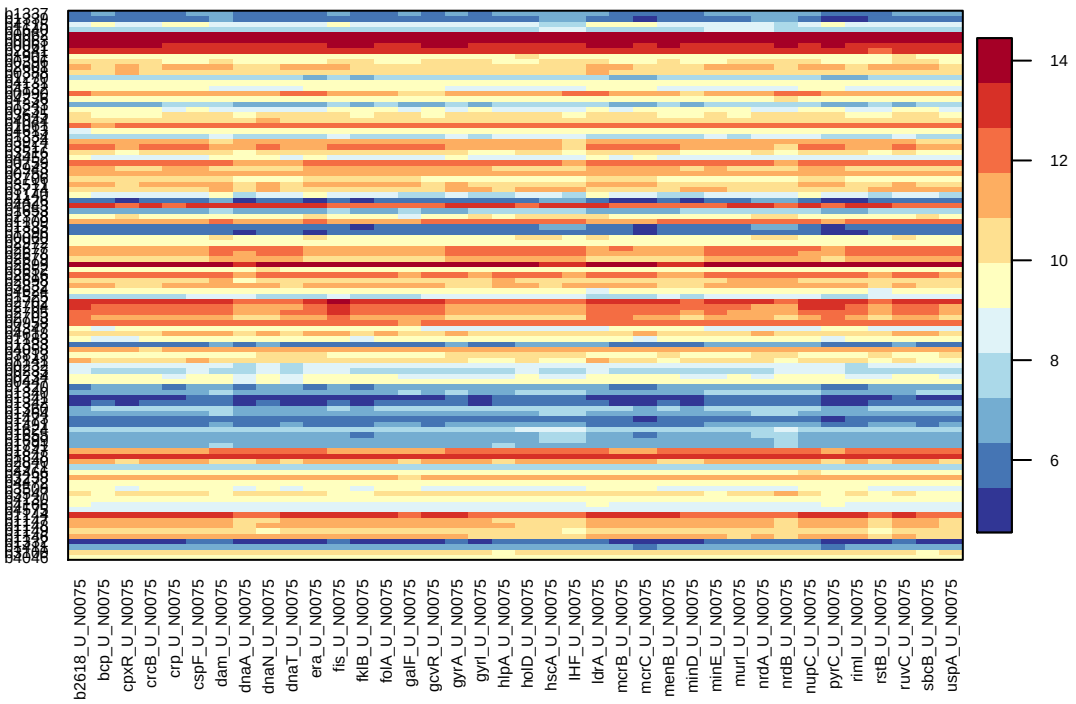

Figure 6: Heatmap for the fifth bicluster identified in the E.coli data. The bicluster consists of 103 genes and 38 conditions

References

Brazma, Alvis, and Jaak Vilo. 2000. "Gene Expression Data Analysis." *FEBS Letters* 480 (1). Elsevier: 17–24.

Cheng, Yizong, and George M Church. 2000. "Biclustering of Expression Data." In *Ismb*, 8:93–103.

Eren, Kemal, Mehmet Deveci, Onur Küçüktunç, and Ümit V. Çatalyürek. 2012. "A Comparative Analysis of Biclustering Algorithms for Gene Expression Data." *Briefings in Bioinformatics*. doi:10.1093/bib/bbs032.

Fehrmann, Rudolf SN, Juha M Karjalainen, Małgorzata Krajewska, Harm-Jan Westra, David Maloney, Anton Simeonov, Tune H Pers, et al. 2015. "Gene Expression Analysis Identifies Global Gene Dosage Sensitivity in Cancer." *Nature Genetics* 47 (2). Nature Publishing Group: 115–25.

Gu, Jiajun, and Jun S Liu. 2008. "Bayesian Biclustering of Gene Expression Data." *BMC Genomics* 9 (Suppl 1). BioMed Central Ltd: S4.

Lazzeroni, Laura, Art Owen, and others. 2002. "Plaid Models for Gene Expression Data." *Statistica Sinica* 12 (1). C/O DR HC HO, INST STATISTICAL SCIENCE, ACADEMIA SINICA, TAIPEI 115, TAIWAN: 61–86.

Li, Guojun, Qin Ma, Haibao Tang, Andrew H Paterson, and Ying Xu. 2009. "QUBIC: A Qualitative Biclustering Algorithm for Analyses of Gene Expression Data." *Nucleic Acids Research* 37 (15). Oxford Univ Press: e101.

Zhou, Fengfeng, Qin Ma, Guojun Li, and Ying Xu. 2012. "QServer: A Biclustering Server for Prediction and Assessment of Co-Expressed Gene Clusters." *PLoS ONE* 7 (3). Public Library of Science: e32660. doi:10.1371/journal.pone.0032660.

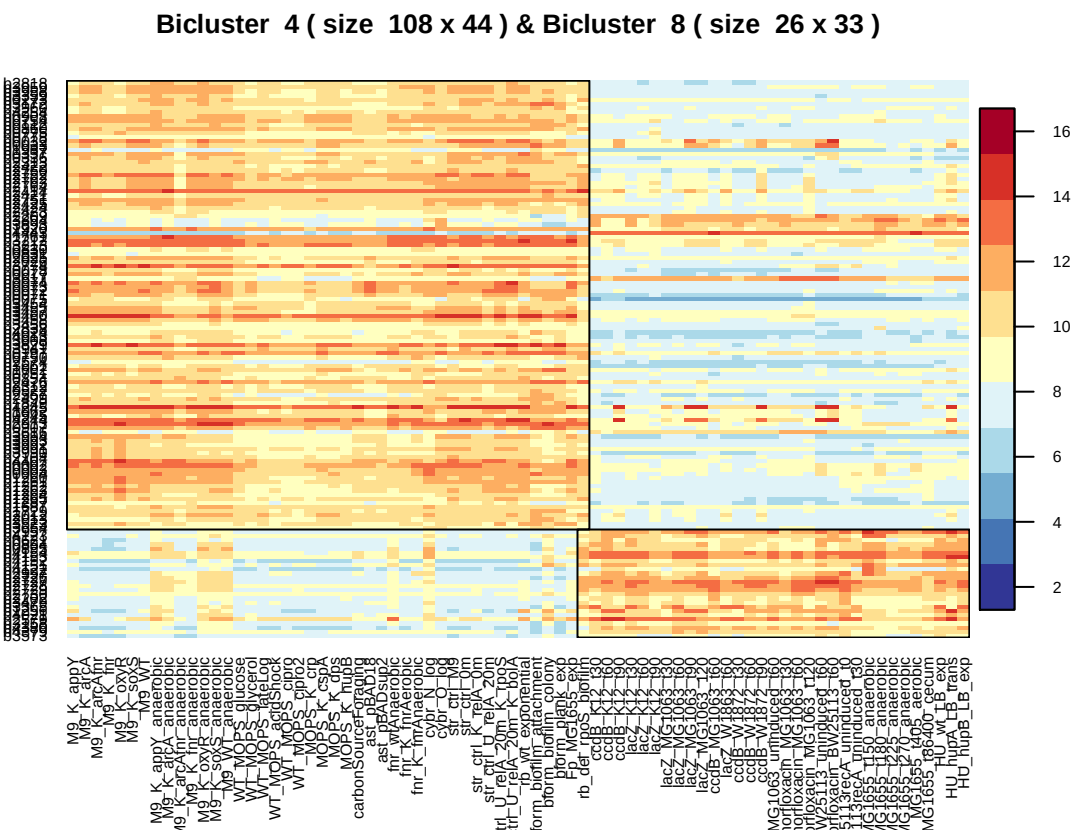

Figure 7: Heatmap for the fourth and eighth biclusters identified in the E.coli data.Bicluster #4 (topleft) consists of 108 genes and 44 conditions, and bicluster #8 (bottom right) consists of 26 genes and 33 conditions

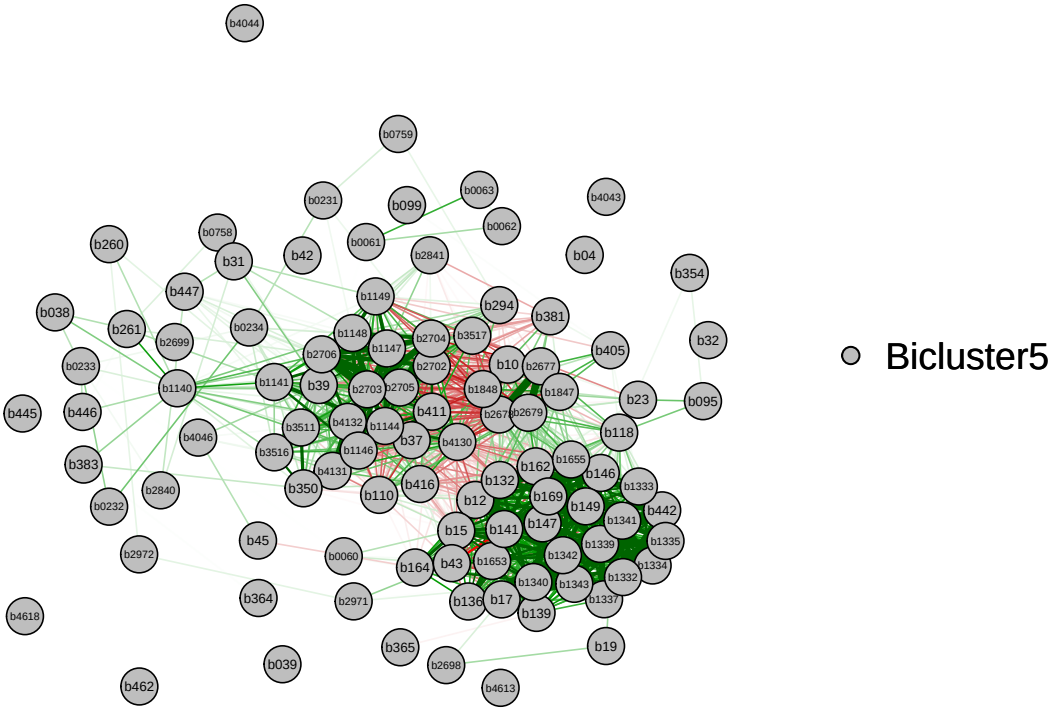

Figure 8: Network for the fifth bicluster identified in the E.coli data.

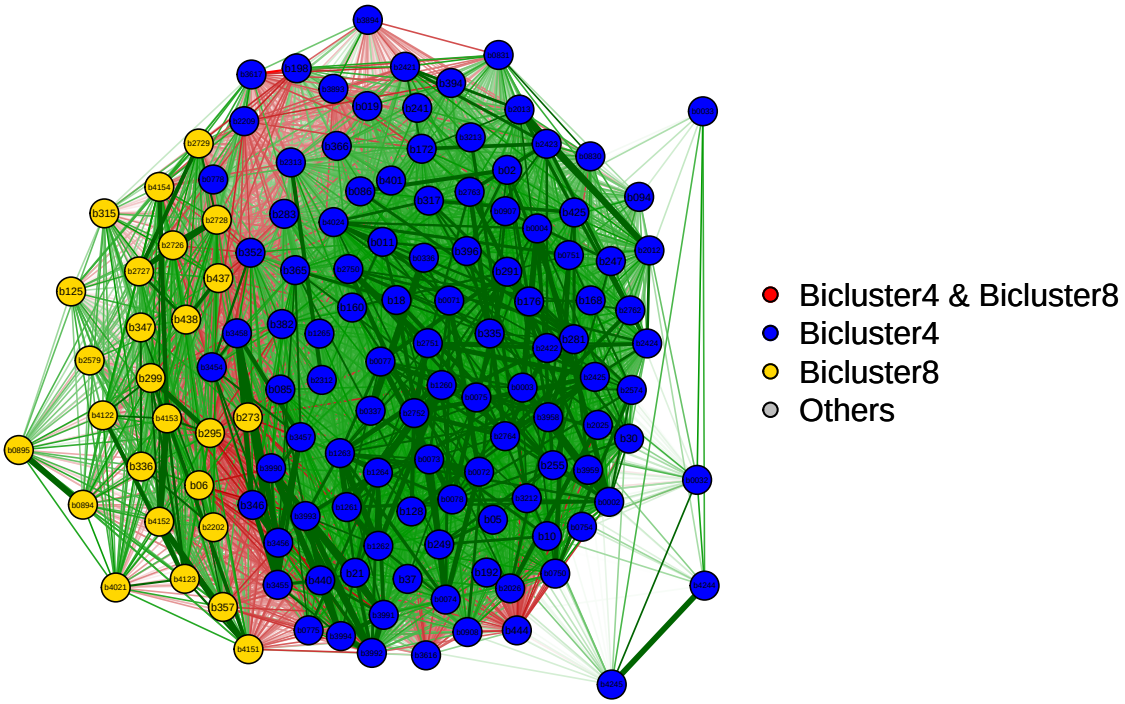

Figure 9: Network for the fourth and eighth biclusters identified in the E.coli data.